Control of Start and Stop events in a HEV

Undergraduate Honors Thesis by

Joseph Paul Porembski

In partial fulfillment of the requirements for

Graduation with Distinction

The Ohio State University
Department of Electrical and Computer Engineering
The Center for Automotive Research

Columbus, Ohio

2007

(Defended: August 22nd, 2007)

Acknowledgements

I would like to thank the following people for their help in the completion of this research.

Dr. Stephen Yurkovich for providing me with this research opportunity and agreeing to be the research advisor on this project. He provided me contact with the capable and intelligent personnel of the Center for Automotive research. His understanding of the research, his experience, his knowledge and his assistance in maintaining a focused goal in light of changing models and parameters were of the utmost importance to the completion of this thesis.

Dr. Marcello Canova for providing me with the tools to complete this control strategy and explaining the greater processes at work in the internal combustion engine. Dr. Canova's ability to help me address problems and his experience were invaluable to my completion of this thesis.

Kris Sevel for connecting me with the greater project of which this research will influence, as well as holding several meetings before leaving the university to assure I was caught up with the Challenge X team.

Dr. Yann Guezennec for taking time from his schedule to be the second faculty examiner for my thesis defense.

Craig Pavlich and the Ohio State University Challenge X team for answering all my questions regarding the mechanics of a Hybrid Electric Vehicle and an Internal Combustion Engine.

Abstract

Hybrid electric vehicles are a key-piece of the future of the automotive industry. It has been demonstrated that HEVs offer benefits to fuel consumption, durability, and drivability to the end-user. The most realized version of the HEV is the belted starter/alternator coupled to the internal combustion engine. Controlling the start and stop events of this engine are paramount to increasing drivability by decreased noise, harshness and vibrations and decreasing fuel consumption.

Model-based control design methods were examined in this research with the goal of creating an optimized controller for implementation on the Challenge X vehicle. The two strategies chosen are apart of two different types of control. The linear quadratic regulator is a type of state-feedback or modern control, and the lead-lag controller is a type of classical control. These two controllers were developed under a linearized model of the engine and belt dynamics which compensated for highly non-linear torque terms by assuming they are negligible or canceling their contribution through a feed-forward term. Through the addition of higher order dynamics, these controllers were verified to be robust, able to handle uncertainties and disturbances during start and stop events.

The controllers were optimized under two different methods, a “static” method which involved developing a cost surface by evaluating a cost function after simulation and calculating the minimum of this surface. The second was a “dynamic” method which involved the calculation of a global minimum via two algorithms, one designed to find the local minima, and the other to define a shrinking space of minima. At the end of the optimization process, the linear quadratic regulator proved to provide the best response.

Table of Contents

List of Figures

Chapter 1: Introduction

1.1 Motivation for this work

A hybrid electric vehicle is a vehicle which combines a conventional propulsion system, such as an internal combustion engine, and an electric propulsion system with a rechargeable energy storage system, such as an electric motor with batteries. As fuel prices increase (Figure 1.1a), market interest in Hybrid Electric Vehicles (HEVs) increases (Figure 1.1b), and popular opinion views it as a segment of the future automotive environment, more and more research effort is spent in the field of HEVs and energy management. This research further separates our economy from dependence on foreign oil, as well as promises better fuel economy and emissions.

These benefits are realized through certain abilities a HEV has which a traditional automobile does not possess. Fuel consumption is minimized by turning off the internal combustion engine during idle and low RPM events, and recapturing energy through regenerative breaking, which converts a portion of the vehicle's kinetic energy into electrical energy during breaking and stop events. These benefits lead to improved durability of the engine as a whole due to elimination of the strain on the internal combustion engine during idle events and braking. The use of HEVs benefits the environment by reducing emissions, caused by reducing fuel consumption, and reducing noise pollution, by using the electric motor at low speeds [1, 2].

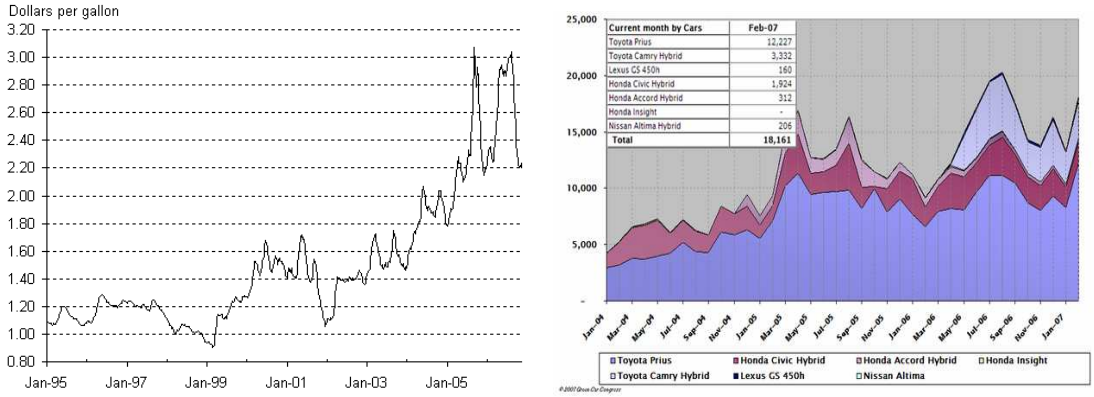

Figure 1.1 - (a) Average price of gasoline in the Continental US from January 1995 to February 2007 (b) Sales of major Hybrid Electric Vehicles per month from January 2004 to February 2007

The HEV configuration in this circumstance is the belted starter/alternator, in which a belt transmission couples the electric motor to an internal combustion engine. This implementation has the benefit of low cost due to the small size required for the proper operation of the electric machine. This configuration has more power than a traditional engine starter; it is able to crank the engine to idle speed in a shorter period of time, resulting in the need for developed start event and stop event control strategies [3], for properly developed control strategies can result in further benefits in fuel consumption and emission in HEVs, as well as overall drivability.

In order for the implementation of a start/stop control strategy to be considered successful, the engine should be able to start and stop with little to no unwanted noise, vibrations, and harshness (NVH). The engine should also be able to start in a reasonably short period of time, without peaking too high compared to the idle speed. These contribute to the overall negative effects on drivability. This will be accomplished through a closed-loop, model-based control strategy. The goal of this research is to accomplish this task.

1.2 Outline of this Thesis

This thesis will explore the development and implementation of a start/stop event control strategy on a Hybrid Electric Vehicle. Chapter 2 will outline the development of this model, explaining the vehicle itself, development of equations, and how the end model for the system was developed. Chapter 3 will cover the initial control strategy developed using strictly first order dynamics, as well as the robustness of these strategies in the face of disturbances and uncertainties. Chapter 4 will present the introduction of higher order dynamics to the control strategy to create a more robust controller. Chapter 5 will finalize the control strategy through two types of optimization.

Chapter 2: Linear and Non-Linear Modeling of a Hybrid Electric Vehicle

2.1 The Challenge X Vehicle and Engine

The Challenge X team at the Ohio State University has developed a series/parallel HEV for a mid-size Sport Utility Vehicle (SUV), the Chevrolet Equinox, replacing the stock engine and transmission with a 1.9l turbo diesel, common rail engine and 6-speed transmission as the Internal Combustion Engine. A KollMorgan 10.6kW permanent magnet electric motor will be used to provide the starter/alternator portion of the HEV's engine [4]. Figure 2.1 details the overall architecture of the Challenge X vehicle, the front portion of the car has the belted starter/alternator coupled to the internal combustion engine (ICE), which will be the focus point of this research.

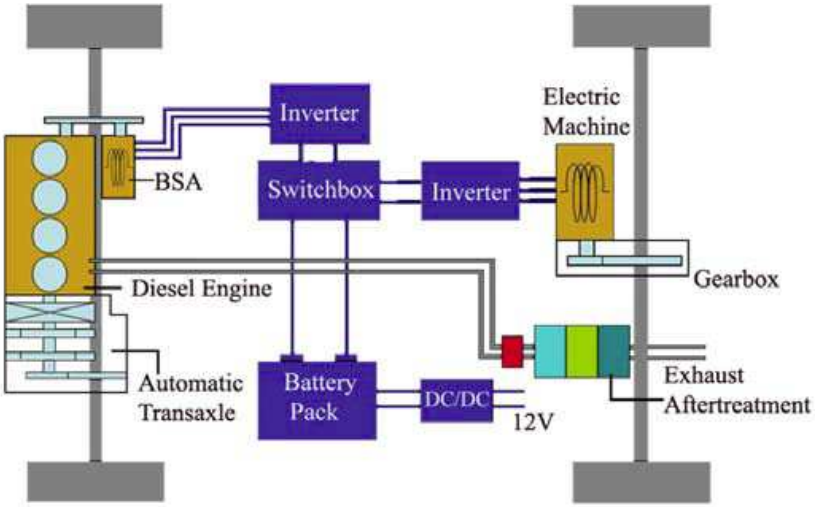

Figure 2.1 - A simplified schematic of the Challenge X vehicle

The basic idea of the control strategy is to control the belted starter/alternator torque (BSA), as a result the "plant" will be the belted starter/alternator coupled with the ICE. Figure 2.2 shows the schematic of the "plant."

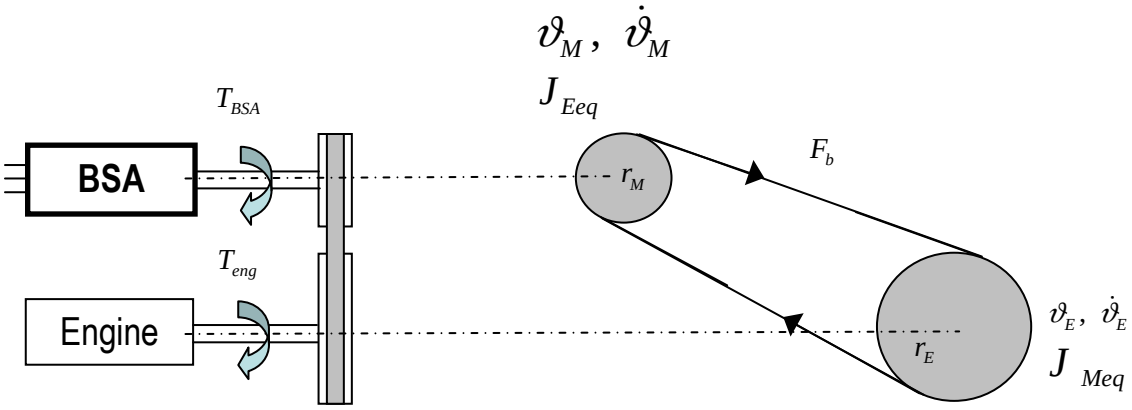

Figure 2.2: Schematic of the belt coupled Engine and BSA

This hybrid system consists of the ICE and BSA mentioned above, the transmission ratio on the belt coupling is 1:1 so the electric motor can deliver the maximum amount of torque during idle conditions, but still be able to match the maximum engine speed [4].

The equations of motion for this system:

$$J_{Eeq}\ddot{\vartheta}_E = T_{eng} + r^2 B(\dot{\vartheta}_M - \dot{\vartheta}_E) + r^2 K(\vartheta_M - \vartheta_E)$$
$$J_{Meq}\ddot{\vartheta}_M = T_{BSA} - r^2 B(\dot{\vartheta}_M - \dot{\vartheta}_E) - r^2 K(\vartheta_M - \vartheta_E) \quad (2.1)$$

In this case ϑ_M and ϑ_E are the angular positions of the BSA and Engine shafts, respectively, and J_{Meq} and J_{Eeq} is the equivalent mass moment of inertias of the BSA and Engine shafts, respectively, and r is the radius of the sprocket. K and B are parameters characteristic of the belt and provided by the manufacturer, representing stiffness and dampening characteristics. The model for the system calculates T_{eng} as:

$$T_{eng} = T_{ind}(\vartheta) + T_m(\vartheta) - T_{fr} + T_{BSA} = J_{Eeq}\ddot{\vartheta}_E \quad (2.2)$$

Where T_{ind} is the indicated torque caused by compression and combustion, T_m is the reciprocating torque, and the most important torque for the purposes of the control strategy, T_{fr} is the friction torque. T_{BSA} is the BSA torque, which will be the control output.

The equations describing the torques involved in the system:

$T_{fr} = k_0 + k_1\omega + k_2\omega^2$ where the k-coefficients are constants defining the contribution of each term of the velocity to the total frictional losses: constant, velocity, and squared velocity.

$$T_{ind}(\vartheta) = RA_P(p - p_{amb})\sin\vartheta\left(1 + \frac{R\cos\vartheta}{\sqrt{L^2 - R^2\sin^2\vartheta}}\right)$$ where p represents the pressure in the cylinder, R is the crank radius, L is the connecting rod length, and A_p is the piston surface area. This torque is phased according to the firing order of the pistons, this term is difficult to account for in model based control, and will have to be handled within the controller differently.

$$T_m = RF_{in}(\vartheta)\sin\vartheta\left(1 + \frac{R\cos\vartheta}{\sqrt{L^2 - R^2\sin^2\vartheta}}\right)$$ where $F_{in}(\vartheta)$ is the inertial force generated by the reciprocating motion of the pistons. Similar to the indicated torque, this term is highly non-linear, and will have to be handled within the controller via an assumption [4].

2.2 Creating a controllable model

For the purposes of creating a controllable model, several assumptions are going to be made. The first is regarding the reciprocating torque, T_M , which experimentally was found to be very small compared to the indicated torque. Since the reciprocating torque also holds highly non-linear behavior, it will be neglected. Figure 2.3 shows the representation of engine torque according to the simplification.

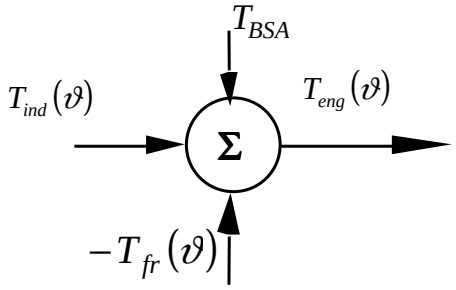

Figure 2.3 Graphical view of engine torque

The next simplification to be made will involve the inducted torque term, T_{ind} . The indicated torque term is highly non-linear, and creates an issue for model based control development. As such, the control action, T_{BSA}, will be split into a feed forward and feedback term. For the purposes of the controller, the torques will be split and defined as follows:

$$T_{eng} = T_{ind}(\vartheta) - T_{fr} + T_{BSA} = T_{ind}(\vartheta) - T_{fr} + (T_{FF} + T_{FB}) = J_{Eeq}\,\dot{\omega}_E$$
$$T_{FF} \approx -T_{ind}(\vartheta)$$

This definition further simplifies the dynamic equation:

$$T_{eng} = T_{FB} - T_{fr} = T_{FB} - \left(k_0 + k_1\omega + k_2\omega^2\right) = J_{Eeq}\,\dot{\omega}_E \qquad (2.3)$$

These two simplifications create a system by which model based control can now be developed. The first necessary action is to linearize the dynamic, non-linear equation. To accomplish this, a set speed has to be chosen: ω_0 and steady state torque $T_{FB,0}$. The perturbations around this point will be defined as: $\delta\omega = \omega_E - \omega_0$ and $\delta T = T_{FB} - T_{FB,0}$. Here the set speed, ω_0, is the idle speed. Approximately 88 radians per second.

Linearization can now be accomplished via the linearization formula:

$$f_L(\omega) = f(\omega_0) + \left.\frac{\partial f(\omega)}{\partial \delta\omega}\right|_{\omega=\omega_0} \delta\omega.$$

With all these pieces the dynamic equation can be linearized:

$$J_{Eeq}\,\dot{\delta\omega} = \delta T + T_{FB,0} - ((k_0 + k_1\omega_0 + k_2\omega_0^2) + (k_1 + 2k_2\omega_0)\delta\omega).$$

In steady state conditions, δT and $\delta\omega$ will both be zero, so to satisfy this condition of the model $T_{FB,0}$ will be defined as: $T_{FB,0} = k_0 + k_1\omega_0 + k_2\omega_0^2$. (2.4)

The linearized, first-order system equation now becomes:

$$J_{Eeq}\,\dot{\delta\omega} = \delta T - (k_1 + 2k_2\omega_0)\delta\omega \quad (2.5)$$

Figure 2.4 is a block diagram of the final control system.

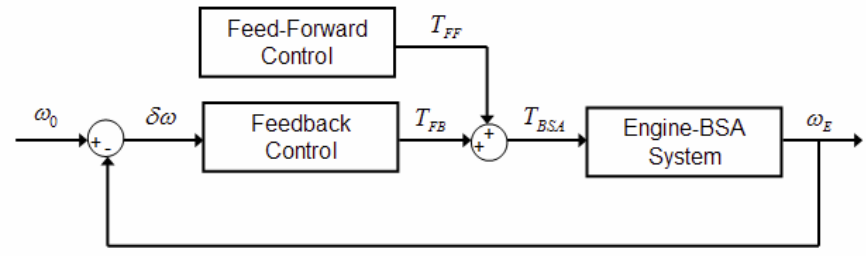

Figure 2.4 Block Diagram of feedback loop

Chapter 3: Creation of a classical and modern control strategy

3.1: Creation of a lead controller

The lead controller design is a type of classical control, specifically a design developed in the Laplace domain. The first step is to convert the unalterable process or "plant" in (2.5) to the Laplace domain:

$$J_{Eeq} s\delta\omega(s) = \delta T(s) - (k_1 + 2k_2\omega_0)\delta\omega(s)$$

From here, this can be put in standard form. The plant has an input of $\delta T(s)$ and an output of $\delta\omega(s)$.

$$[J_{Eeq} s + (k_1 + 2k_2\omega_0)]\delta\omega(s) = \delta T(s)$$

$$G(s) = \frac{\delta\omega(s)}{\delta T(s)} = \frac{1}{J_{Eeq} s + k_1 + 2k_2\omega_0} \qquad (3.1)$$

For notation purposes $(k_1 + 2k_2\omega_0) \equiv A(\omega_0)$. Now the plant is a first order transfer function with a zero at $\frac{-A(\omega_0)}{J_{Eeq}}$, which depending on the set point is between 0 and negative 1. If the set speed is defined as 850 rotations per minute (rpm), the plant has a zero at -0.9220.

Due to the nature of the plant, with the zero close to the origin, a lead compensator will be used to shift the root locus further into the left half plane and improve the systems overall response time. The general form of a lead compensator:

$G_c(s) = K\frac{s-z}{s-p}$, where z < p. For the purposes of analyzing the behavior of the lead controller the form will be redefined: $G_c(s) = K\frac{s-z}{s-\alpha z}$, where α > 1.

To accomplish the design of this controller, the engine should be able to start quickly, in a few tenths of a second, should have minimal overshoot, and should limit NVH for the purpose of drivability. NVH will be measured by RMS acceleration of the body of the engine, which causes vibrations and harshness. The other concern will be power demand of the engine, the engine should be able to accomplish start and stop with minimal power demand for the concerns of conservation and the fact that the BSA torque is limited to 82 Nm, and of course zero steady state error.

In the case of this system, provided both correctness of the model and the absence of disturbances, steady state error should be zero as per the definition of our steady state conditions in (2.2). The primary focus to begin the design will be rise time, which optimally should be around 0.4 seconds.

In this case, the zero of the compensator will be placed on the pole of the plant, so the problem then reduces to finding a combination of α and k such that the necessary conditions on the controller will be satisfied. With that placement, the damping ratio will be 1 as it is strictly a first order system, so the use of an exponential equation can determine the necessary location of the closed loop pole to achieve the necessary rise time.

$N(t) = C_1 e^{-\lambda_1 t}$, where $\lambda_1 = \frac{1}{\tau}$. The settling time can be defined as 4τ or 4 time constants, so the necessary location of λ_1:

$$4\tau = 0.4$$
$$\tau = 0.1$$
$$\lambda_1 = \tfrac{1}{\tau} = 10$$

This means the closed loop pole will have to be less than or equal to -10 in order for the desired rise time to be achieved [5]. The next step is to define the parameters of the lead controller such that they fit within specifications, once an α is defined, k can be chosen such that the closed loop pole is placed in the necessary location. Figure 3.1 details simulation results as α is varied.

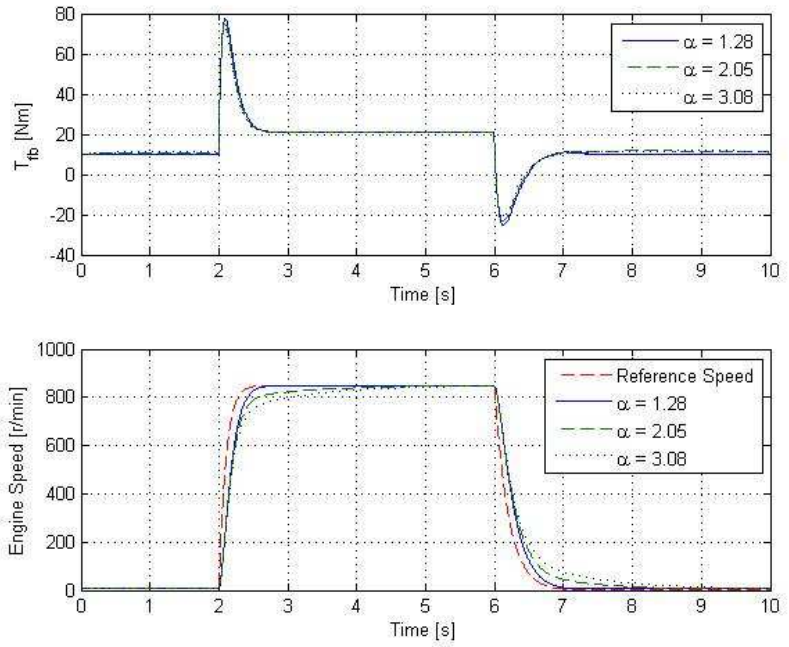

Figure 3.1 - Simulation results with varying α

Figure 3.1 shows the results of simulation in the linearized model as α is varied, with a constant k. The top plot in this figure is of the control action, T_{FB}, and the bottom plot is of engine speed, ω_E, in rotations per minute. The red-dashed line is the reference speed that the controller is tracking. It is clear that for a given k; an α close to but slightly greater than 1 achieves the best response. The chosen α will be 1.28, which will place the pole of the compensator at -1. Figure 3.2 details the root locus plot of the compensator and plant, displaying the k necessary to achieve the desired response.

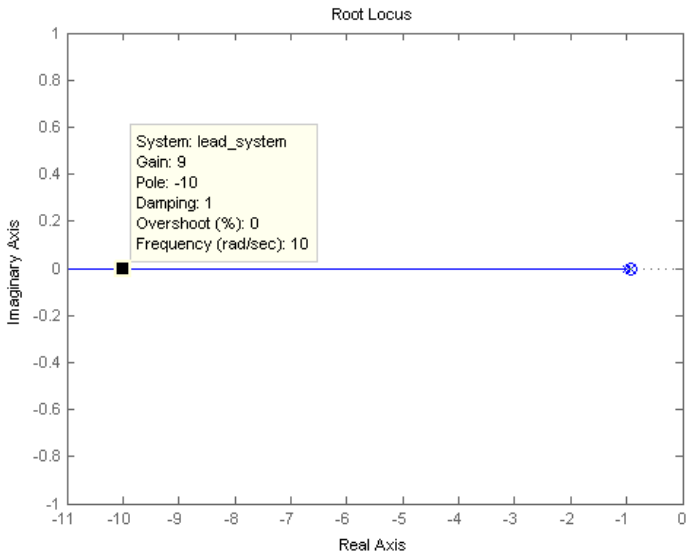

Figure 3.2 - Root Locus plot of the lead compensated system

The final form of the lead compensator:

$$G_C(s) = 9\frac{s + \dfrac{A(\omega_0)}{J_{Eeq}}}{s + 1.28(\dfrac{A(\omega_0)}{J_{Eeq}})}. \qquad (3.2)$$

3.2 Creation of a Linear Quadratic Regulator

The Linear Quadratic Regulator is a type of modern controller which uses a state space representation of a system in controller canonical form, seen in (3.3), and using two cost vectors, Q and R, minimizes a cost function, seen in (3.4) [5].

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{3.3}$$

$$J = \int_0^\infty (x^T Q x + u^T R u)\,dt \tag{3.4}$$

$$u = -K \bullet x \tag{3.5}$$

The result of calculations in a linear quadratic regulator is the vector K, which is fed back in through the input, as defined in (3.5), this state feedback is what creates the feedback loop. The first step to developing this controller is taking the system equation or "plant" and converting it into controller canonical form. The task of converting this equation to the Laplace domain has already been done in (3.1), so a simple conversion can be done using the single input, first order plant transfer function.

$$G(s) = \frac{1}{J_{Eeq} s + A(\omega_0)} = C(sI - A)^{-1} B + D \tag{3.6}$$

A couple terms in the controller canonical form are already known, it is going to be assumed that D = 0 (for this model) and that B = 1. The remainder of the calculations are simple: $A = \left[\frac{-A(\omega_0)}{J_{Eeq}} \right]$ and $C = \frac{1}{J_{Eeq}}$. The last piece that must be specified is the actual states: $x = [\delta\omega]$, $y = [\omega]$, and $u = [\delta T]$.

With the pieces of the controller canonical form specified, the Linear Quadratic Regulator (LQR) becomes a problem of specifying values of Q and R to achieve the best response. Figures 3.3 details the response of the linearized model as Q is varied with R fixed at 1. Figure 3.4 details the response of the linearized model as R is varied with Q fixed at 1. The final K chosen was 2.372.

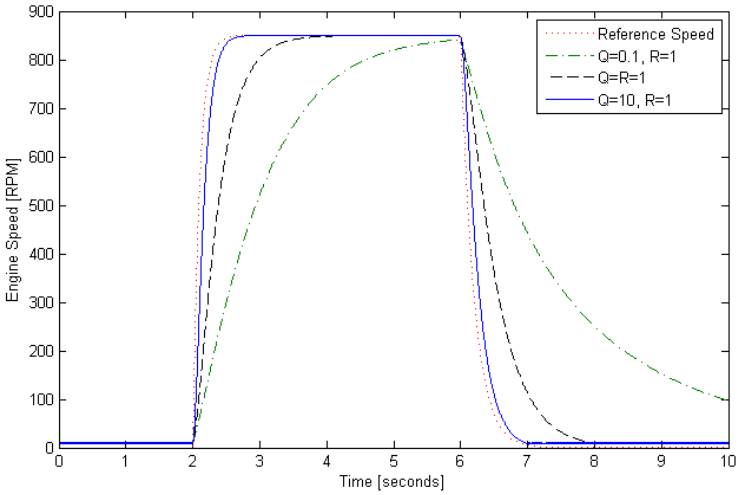

Figure 3.3 – Simulation results of Engine RPM as Q varies

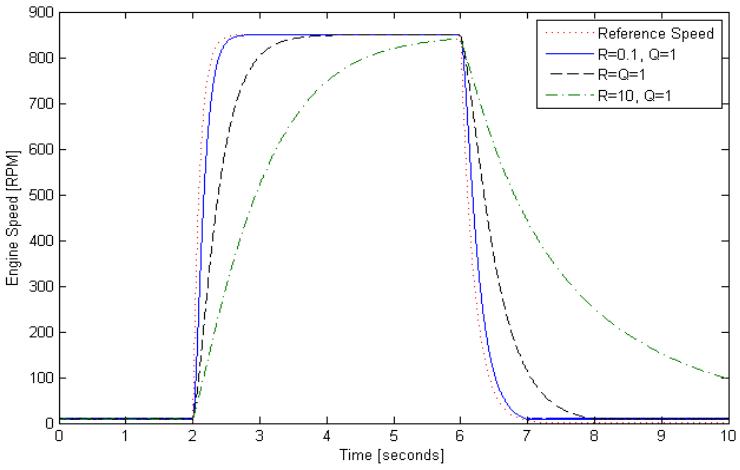

Figure 3.4 - Simulation results of Engine RPM as R varies

3.3 Issues of Robustness

Typical disturbances could be bias, numerical errors in or complete absence of the feed forward term, errors in the frictional coefficients used to estimate the steady-state term, or user introduced disturbances during start and stop events. A model based design approach was used to develop the controllers in this chapter assuming ideal conditions, the problem now becomes how these controllers react to disturbances and uncertainties, and how these issues can be corrected.

The first concern is how each controller responds without the inclusion of the feed forward term.

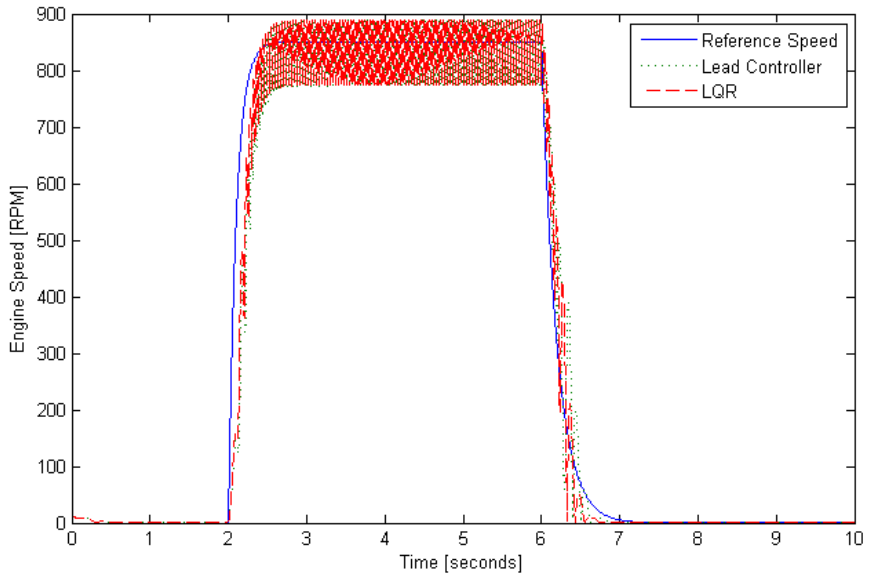

Figure 3.5 - Lead controller and LQR without the inclusion of the Feed Forward term

Figure 3.5 is the linear simulation results of both controllers without the feed forward term. The general behavior of the steady state behavior is almost satisfied, the average steady state error is approximately zero, but the oscillations caused by the absence of the feed forward term and the sharp stop event hinder the drivability of the

vehicle. This represents the worst possible circumstance of the start/stop of a vehicle when disturbances on the feed forward term are considered.

Figure 3.6 is the linear simulation results of the lead controller under various disturbances that could come on the feed forward term. The saturation and bias represent a numerical error in the formula or the estimation of the magnitude of the feed forward term. The disturbance could represent a timing issue in the start/stop event or a user-introduced error. Figure 3.7 is the linear simulation results of the LQR controller under the same disturbances.

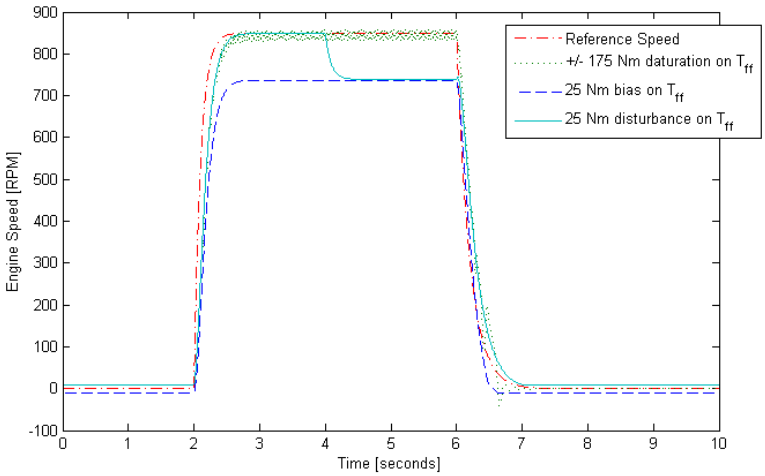

Figure 3.6 - Simulation results of the lead controller under various disturbances

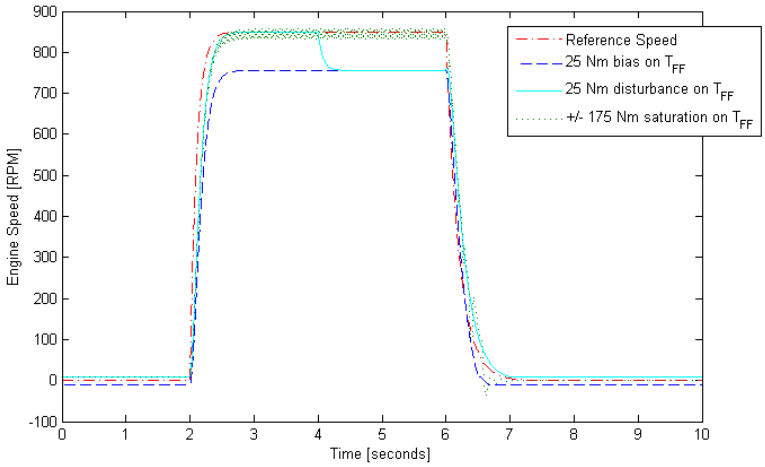

Figure 3.7 - Simulation results of the LQR controller under various disturbances

Figure 3.8 is the linear simulation results of the lead controller in the presence of uncertainties in the frictional coefficient terms. These uncertainties could be caused by cold or hot starts, as these terms have some dependence on temperature, as well as changes in other characteristics of the engine and human error in derivation. Figure 3.9 is the linear simulation results of the LQR controller under the same uncertainties in the frictional coefficient terms.

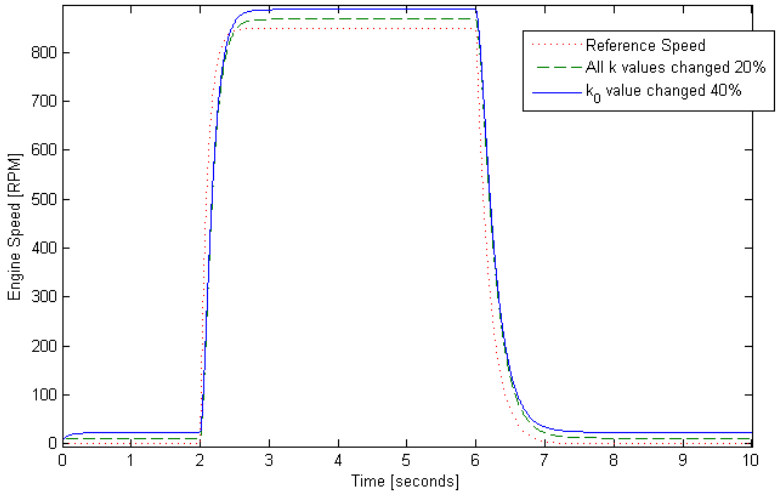

Figure 3.8 - Simulation results of the lead controller under uncertainties in friction constants

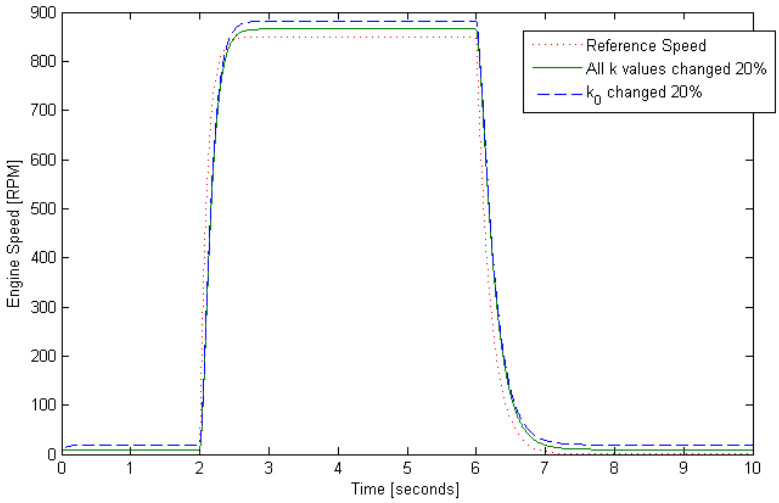

Figure 3.9 - Simulation results of the LQR controller under uncertainties in friction constants

Chapter 4: Adding dynamics to the control strategy

4.1 Creation of a lead-lag controller

A lead-lag controller or network is a type of classical control; it builds on the lead controller in chapter 3. Using the same plant as the lead controller, it adds higher order dynamics to the control strategy for the purpose of compensating of disturbances and uncertainties. The general form of a lead-lag controller or network:

$$G_C(s) = K\frac{(s+z_1)(s+z_2)}{(s+\alpha_1 z_1)(s+\alpha_2 z_2)} \qquad (4.1)$$

Where typically $|z_1| > |z_2|$, $\alpha_1 > 1$, and $\alpha_2 < 1$, the important component of this controller is that there are two sets of pole/zeros, one a lead network where the zero is closer to the origin than the pole, and the other a lag network where the pole is closer to the origin than the zero [5]. The first step in developing this controller will be to set $z_2 = \frac{A(\omega_0)}{J_{Eeq}}$. The second step is important for handling the robustness issues addressed in chapter 3, which is to set $p_2 = \alpha_2 z_2 = 0$, or introduce an integrator to the controller. The disturbances and uncertainties introduced in chapter 3 all caused constant or step errors, so introducing a single integrator to the controller will result in zero steady-state error in the presence of all the reasonable disturbances and uncertainties, and hinder the oscillations caused by saturation in the feed forward term. The combination of this pole/zero set satisfies the lag portion of the controller. From here $z_1 \equiv 12.5$, this will shape the root locus so that as K increases, the closed loop poles of the system will move beyond the necessary natural frequency to attain the required rise time by one time constant.

Choosing the same α of 1.28, this means $p_1 = \alpha_1 z_1 = 1.28 \bullet 12.5 = 16$, keeping these specifications a K needs to be chosen such that the specification are met. Figure 4.1 is the root locus plot of the lead-lag controller and plant, from this plot a K≥ 4.5 will satisfy the specifications. This results in the lead-lag controller:

$$G_C(s) = 4.5\frac{(s+12.5)(J_{Eeq}s + A(\omega_0))}{s(s+16)}. \qquad (4.2)$$

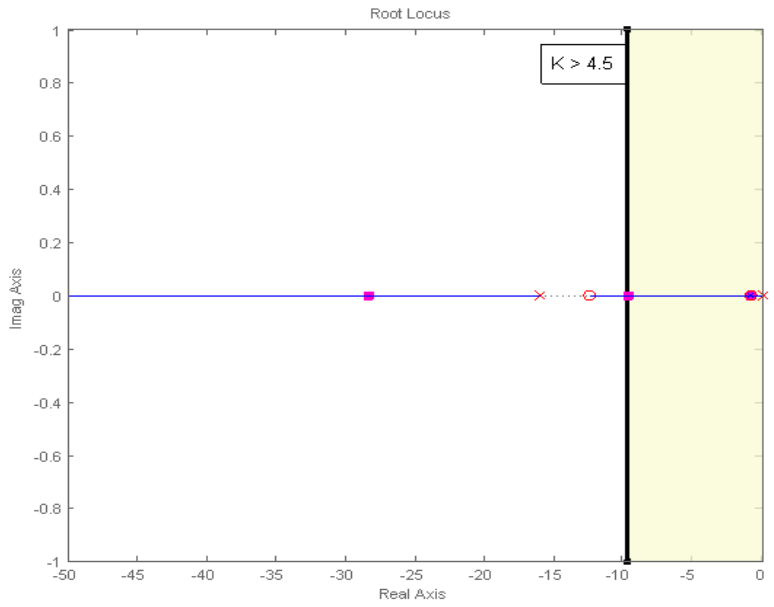

Figure 4.1 - Root locus plot of lead-lag compensated system

With these dynamics added, the disturbances and uncertainties of chapter 3 are handled to still provide an acceptable response. Figure 4.2 shows the response of the lead-lag controller in the presence of the various disturbances of chapter 3: saturation and bias of the feed forward term representing numerical errors, and a disturbance representing a potential user-error. As designed, the disturbances are compensated for and the response is within specifications.

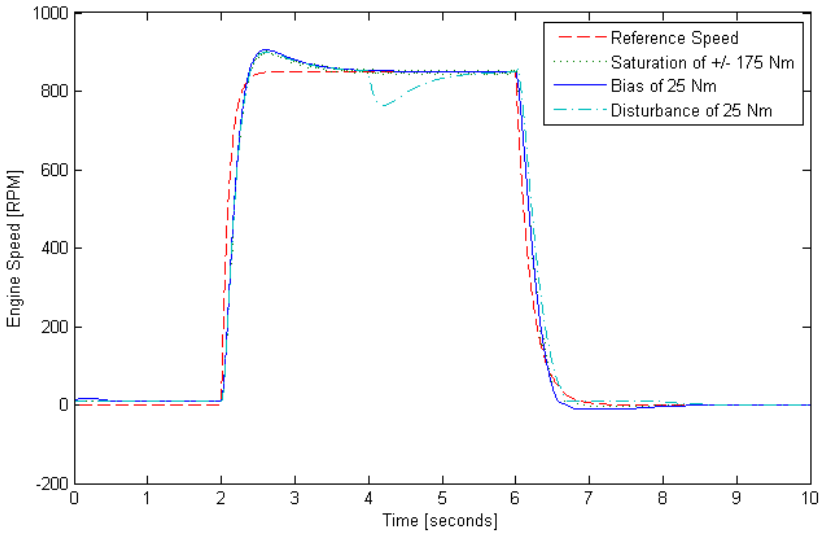

Figure 4.2 – Simulation results of the lead-lag controller under various disturbances

Figure 4.3 details the response of the lead-lag controller in the presence of uncertainties in the friction coefficients. As expected the uncertainties, which could be caused by temperature changes, varying conditions in the engine, and human error in derivation, are compensated for and the response is within specifications.

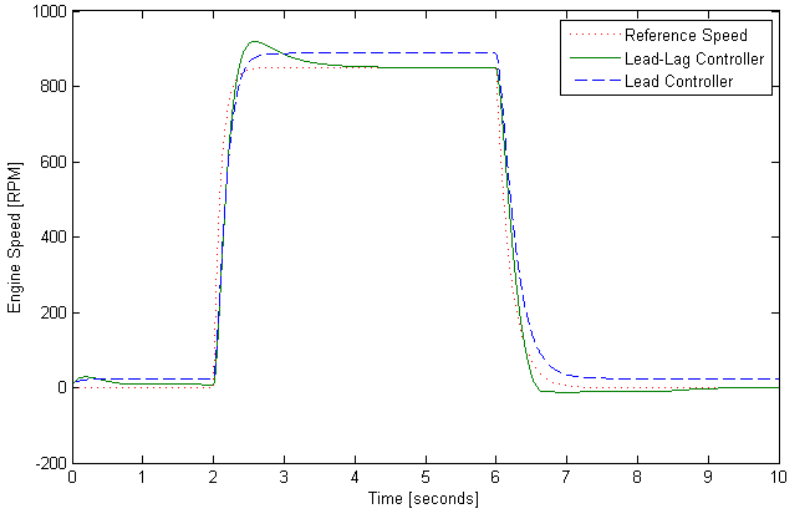

Figure 4.3 - Simulation results of the lead-lag controller under uncertainties in friction constants

Unfortunately complete elimination of the feed forward term is not completely handled by this lead-lag controller. Due to the assumption made on the feed forward term, no feasible controller can compensate for its complete absence. Figure 4.4 demonstrates the response of the linear model to the elimination of the feed forward term with lead-lag compensation. The steady state error is handled completely; the oscillations and sharp stop event create NVH which hinders drivability.

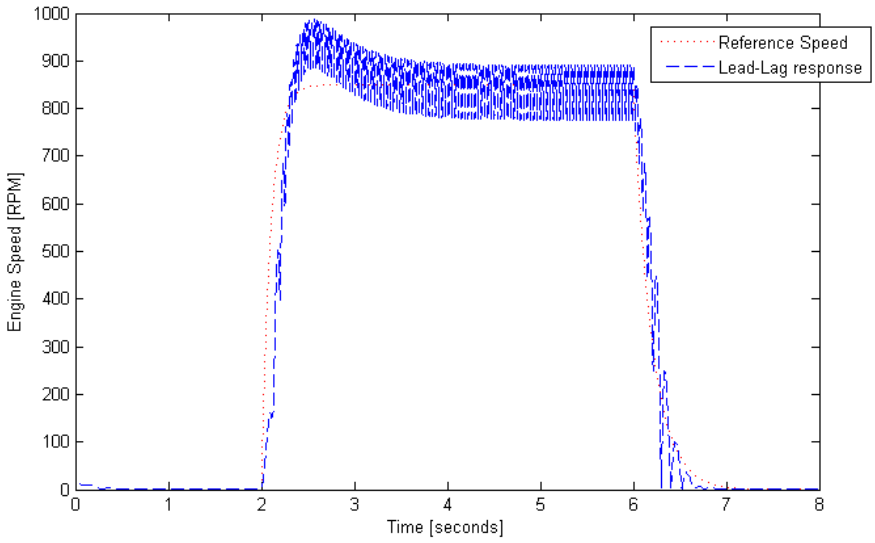

Figure 4.4 - Lead-lag controller without the inclusion of the Feed Forward term

4.2 Creation of a Linear Quadratic Regulator with integrator

The Linear Quadratic Regulator (LQR) controller developed in chapter 3 suffered in the face of disturbances and uncertainty, much like its classical counterpart the lead controller. The beginning of chapter 4 demonstrated that the introduction of higher order dynamics to the controller can compensate for these real disturbances and uncertainties. In the classical control area, an integrator was added as apart of a lag element in a lead-lag controller to achieve zero steady-state error and compensate for disturbances and uncertainties. The same principle can be applied to the linear quadratic regulator with the addition of a new state.

The input and output vectors will obviously remain the same, the basics of the controller cannot be changed. The state vector will be manipulated to include an integrator state. This state will implement the function:

$$\dot{\delta\omega}_I = y + \delta\omega_I = C \bullet \delta\omega + \delta\omega_I \qquad (4.3)$$

This is a basic integrator state where the next state is the output plus the previous state. This will alter the state vector $x = \begin{bmatrix} \delta\omega \\ \delta\omega_I \end{bmatrix}$, as well as A and B of the controller canonical form.

$$A = \begin{bmatrix} -\frac{A(\omega_0)}{J_{Eeq}} & 0 \\ \frac{1}{J_{Eeq}} & 1 \end{bmatrix}; B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

The integrator state in this case is not influenced by the input, but does influence the output via the state vector as defined in (3.5).

The cases of K chosen were the same as those in chapter 3. This resulted in $K = [2.505 \quad 0.703]$. Figure 4.5 shows the linear model results of the higher order LQR to disturbances and uncertainties in the feed forward term: saturation and bias of the feed forward term representing numerical errors, and a disturbance representing a potential user-error. As anticipated the LQR responds well within specifications.

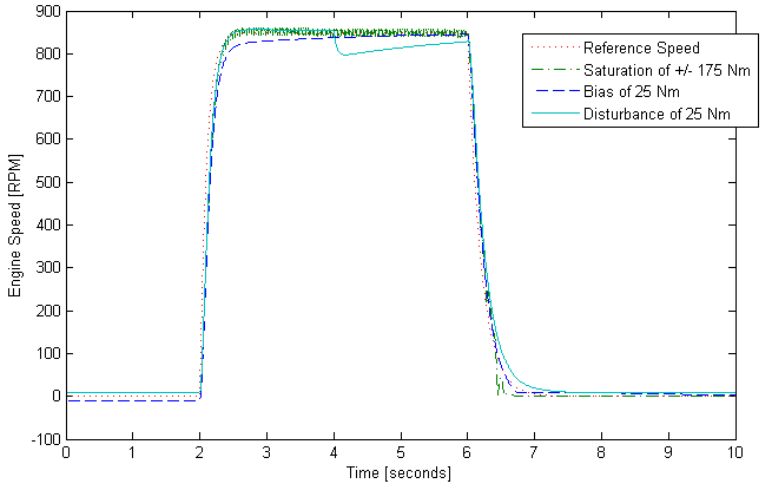

Figure 4.5 - Simulation results of the higher order LQR controller under various disturbances

Figure 4.6 displays the results of the linear model simulation of the higher order LQR under uncertainties in the friction coefficients, caused by temperature changes, varying conditions in the engine, and human error in derivation. These uncertainties are compensated for, and the response is within specifications.

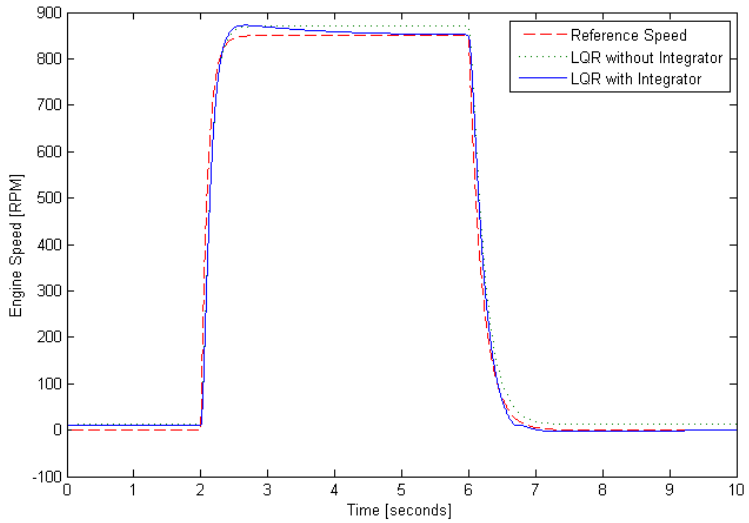

Figure 4.6 - Simulation results of the higher order LQR controller under uncertainties in friction constants

Figure 4.7 shows the simulation results of the LQR with integrator when the feed forward term is removed. As with previous controllers, the feed forward term is too complex to be compensated for by adding higher orders.

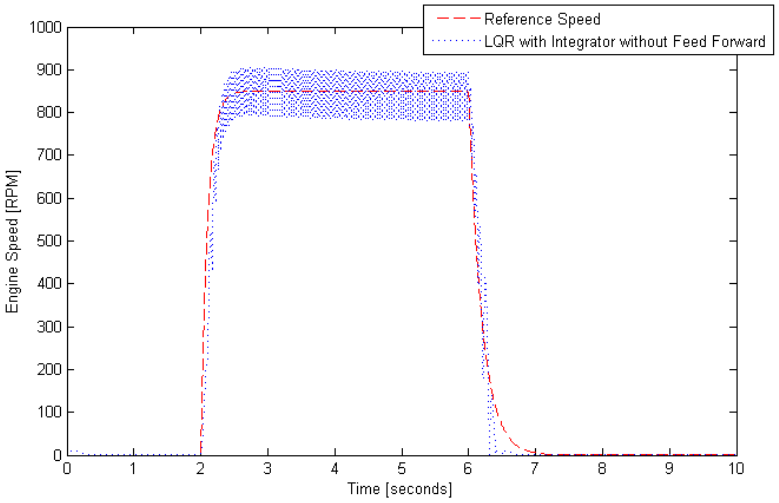

Figure 4.7 - Higher order LQR controller without the inclusion of the Feed Forward term

4.3 Verification on a non-linear model

For the purposes of verifying the research conducted, two models were developed. To this point, the linearized model has been used. This model was developed specifically control development and takes into consideration many of the assumptions made in chapter 2. A full non-linear model was developed [7], it considers the saturation of the torque the controller can supply and ignores some of the assumptions made in chapter 2. This model provides a better simulation of actual experimental engine starts.

Chapters 3 and 4 dealt extensively with development of two separate controllers which could accommodate real conditions within an engine. Figure 4.8 demonstrates an engine start with each controller in a non-linear environment. The response is more of what is expected of a car engine.

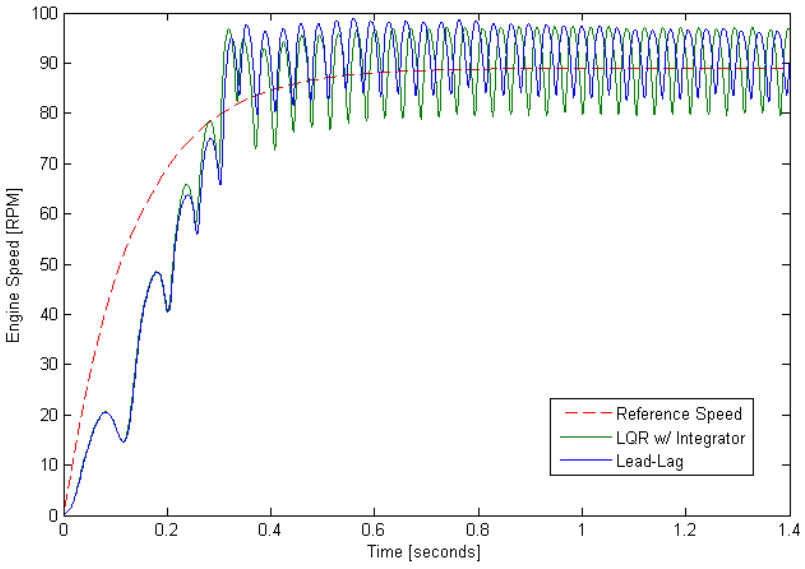

Figure 4.8 - Non-linear simulation of LQR and Lead-Lag

Chapter 5: Optimization

5.1 Static Optimization

There are two types of optimization considered for finalization of the controller to end up on the Challenge X vehicle. The first is a "static" optimization. The non-linear simulation will be ran at different points along a plane, each point representing a set of conditions the controller can run under. The Lead-lag controller will have alpha and K, and the LQR will have Q and R. Once the simulation is ran, a cost function will be evaluated and placed on the map. The optimal choice is simply the minimum of this map.

There are four parameters of interest in the calculation of the cost of a particular controller. Percent overshoot, defined as $PO = \frac{\max(\omega) - \omega_0}{\omega_0} \times 100$, rise time, which will be defined as the time at which the engine speed first reaches ω_0; power demand, defined as the $\max(\omega_E \cdot T_{BSA})$, and the RMS acceleration, defined as $\sqrt{\sum \frac{\dot{\omega}_E \cdot \dot{\omega}_E}{t}}$.

The cost function will be simple:

$$C(l) = \begin{bmatrix} C_{PO} & C_r & C_P & C_{RMS} \end{bmatrix} \cdot \begin{bmatrix} PO \\ t_r \\ P \\ a_{RMS} \end{bmatrix} \tag{5.1}$$

C will be defined as $\begin{bmatrix} 0.4 & 0.4 & 0.1 & 0.1 \end{bmatrix}$. Figure 5.1 demonstrates the static cost plot of the lead-lag controller with α and K as the items to be optimized.

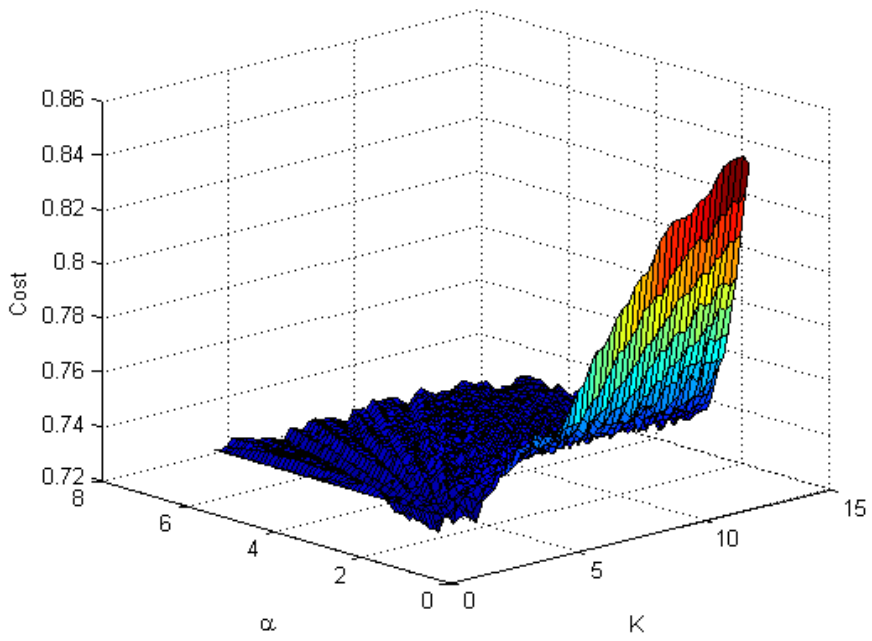

Figure 5.1 - Plot of the lead-lag controller cost

The minimum of this surface plot occurs at an α=1.81 and K=3.25, this would specify the static-optimized lead-lag controller as:

$$G_C(s) = 3.25\frac{(J_{Eeq}s + A(\omega_0)(s+12.5)}{s(s+22.63)} \tag{5.2}$$

Figure 5.2 shows the cost plot of the LQR controller with a varying R and constant Q, Figure 5.3 shows the cost plot of the LQR controller with a varying Q and R constant at 987.

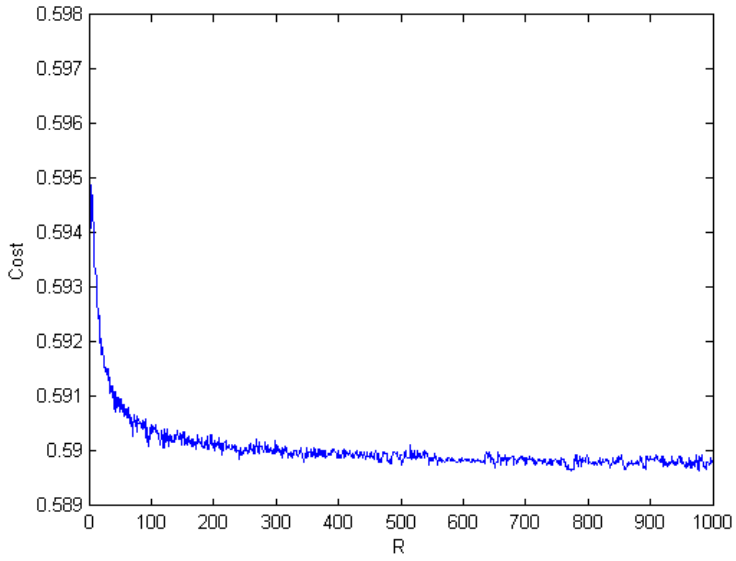

Figure 5.2 - Plot of the LQR controller cost versus R

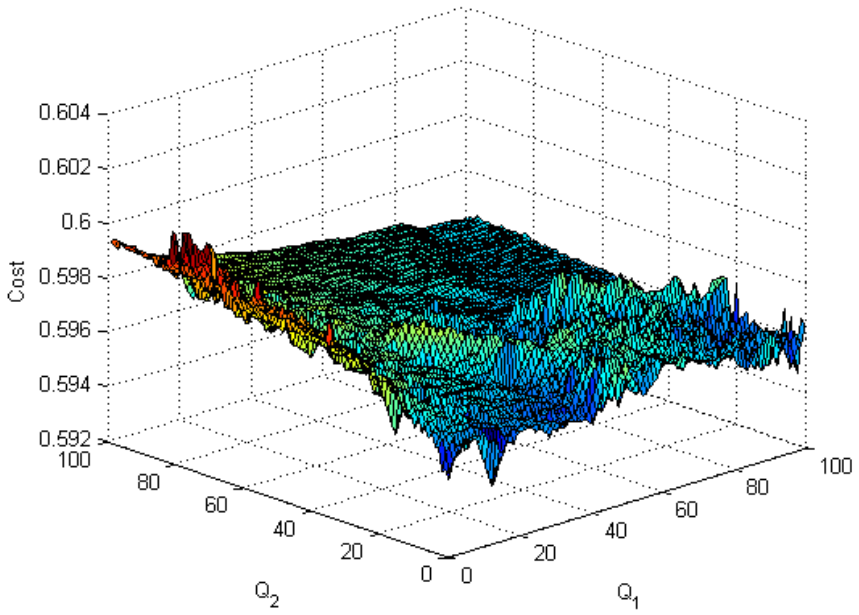

Figure 5.3 - Plot of the LQR controller cost versus Q

The minimum of the surface plot occurs at $Q = \begin{bmatrix} 10.25 & 0 \\ 0 & 1.05 \end{bmatrix}$, this yields

$K = \begin{bmatrix} 2.203 & 0.7204 \end{bmatrix}$.

Its important to note that due to the nature of the cost parameters; the plots in figures 5.1 and 5.3 have elements of noise to them. Percent overshoot, power demand, and rise time are all point-defined values as opposed to integral or averaged metrics, so while these surfaces have several peaks and valleys, their minima might not necessarily reflect an exact minima. Instead they present more of a general idea of the behavior of the cost function as parameters are changed. The minima still provide a good basis for comparison.

5.2 Dynamic Optimization

The second type of optimization is a “dynamic” optimization or unconstrained nonlinear optimization. The cost function defined in this chapter is a scalar valued function with a vector of inputs. The first assumption made is that this function value set is not convex, there are several local minima and maxima, and the goal of this optimization is to find the global minima. The algorithm used to find a minimum is to provide a starting point where the process will begin, around this starting point a simplex is created, which is a series of points forming a shape (triangle, pyramid, etc) around the starting point. A point is chosen at or around each corner point, and parameters are changed to pull the corner point in a set direction which lowers the value of the function. This is repeated until eventually each corner point of the simplex converges on the same minimum.

There are several pre-defined processes which accomplish this first algorithm of finding a minimum given a starting point. With the capability of finding local minima, the problem becomes using this smaller algorithm to find global minima for a given cost function. To accomplish this a larger function will be defined around this algorithm, a set sub-space will be defined $S \in \Re^n$, in this sub-space a set of points will be defined along each axis such that any point $(x, y...) \in S$. Each point within the sub-space will be run through the smaller algorithm to generate a local minimum; this set of local minima will redefine another, smaller sub-space, which will be ran through the larger algorithm again until eventually S converges on 1 element, the global minimum. If this convergence never takes place, S is output for the user to determine or estimate the global minimum.

The use of computer programs, namely MATLAB, was used heavily due to the involved and recursive calculations. Appendix A contains the MATLAB code used to find the global minimum.

The results of this function with the previously defined cost vector for the lead-lag controller:

$$G_C(s) = 0.811\frac{(J_{Eeq}s + A(\omega_0))(s + 54.838)}{s(s + 100.762)} \tag{5.3}$$

The results of this function with the previously defined cost vector for the LQR controller:

$$\begin{aligned} Q &= \begin{bmatrix} 23.90 & 0 \\ 0 & 1.023 \end{bmatrix} \\ R &= 297.399 \\ K &= \begin{bmatrix} 2.068 & 0.7392 \end{bmatrix} \end{aligned} \tag{5.4}$$

5.3 Simulation results and comparisons

Now a final, optimized controller has to be chosen for implementation on the Challenge X vehicle. In both cases the controllers that came out of the dynamic optimization result in lower cost controllers. Figure 5.4 compares the two controllers on engine start and Figure 5.5 compares the two controllers control energy requirements.

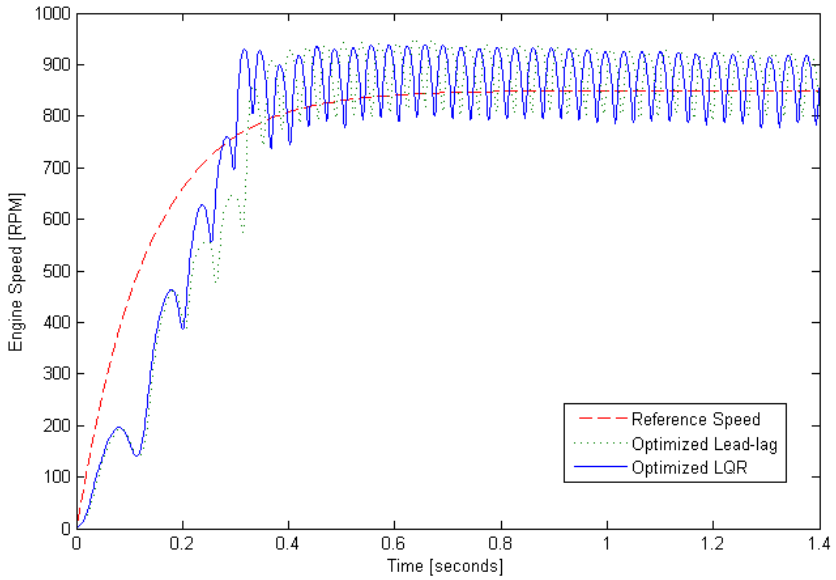

Figure 5.4 - Optimized LQR vs. Optimized Lead-lag engine start

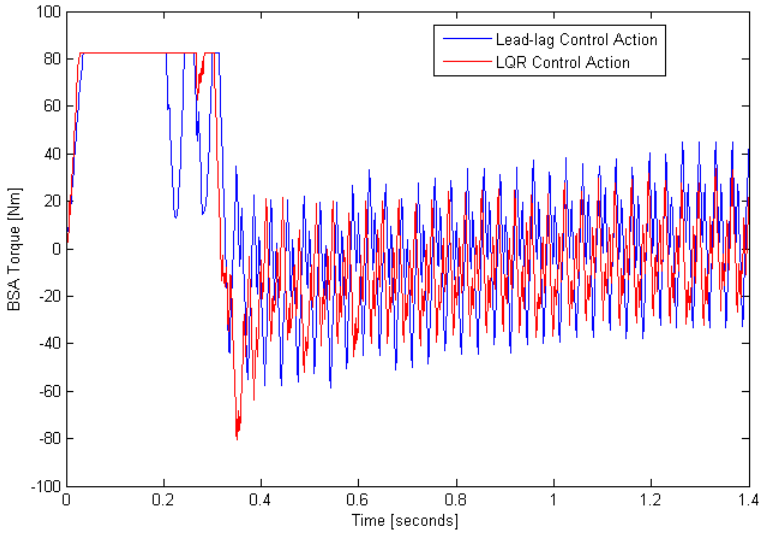

Figure 5.5 - Optimized LQR vs. Optimized Lead-lag BSA Torque

Conclusions

Hybrid electric vehicles are an integral part of the future of the automotive industry; the Challenge X competition and vehicle are prime examples of this. They offer benefits through decreased fuel consumption and emissions, recycling of energy within the engine to keep the electric motor charged, increased durability from the reduced stress on the internal combustion engine, and reduced noise pollution from the quieter running electric motor during idle events. These benefits offer decreased dependency foreign oil and offer savings and better drivability to the end-user.

The configuration of the engine in a Hybrid electric vehicle allows for the control of start and stop events. Control of these events delivers even greater benefits to the end-user in drivability, fuel economy, and emissions. Control can be accomplished through a model-based approach which takes advantage of linearization of a simplified model. This simplified model ignores one negligible, non-linear torque, the reciprocating torque, and compensates for a non-negligible, non-linear torque, the indicated torque, through a feed-forward term.

With a model in place, two types of control strategies are developed and compared: a state-feedback or modern control method, linear quadratic regulator, and a transfer-function or classical control method, the lead-lag controller. The controllers were developed and designed to be robust, able to handle uncertainties and disturbances in the model and experimentally on the engine.

Multiple optimization processes were developed for the control strategies, so that the final controller implemented on the vehicle is assured to be the most optimal. The first is a “static” optimization process which used a plane where each axis represented a

parameter of each controller. Points on these planes were simulated and a cost value for the simulation was computed based on the percent overshoot, rise time, power demand, and RMS acceleration. The minimum of this final surface was defined to be the optimized controller. The second is a “dynamic” optimization process. It takes advantage of a recursive algorithm by which a space is defined of parameters of each controller and a smaller algorithm takes each of these points within in this space and computes a local minima. The larger algorithm is run again on this space of minima until eventually the space converges on one point, the global minimum.

The goal of this research was to develop an optimal, model-based controller for implementation on the Challenge X vehicle. After the completion of the research, the optimized Linear Quadratic Regulator gives the best overall response at the best cost. The advantage of the Lead-lag control strategy is consistency in response; there are large regions around the optimized parameters which provide a similar response.

References

[1] Internet: http://en.wikipedia.org/wiki/Hybrid_electric_vehicle. Accessed: 08/16/07

[2] Internet: http://www.bts.gov/publications/white_house_economic_statistics_briefing_room/november_2006/html/highway_retail_gasoline_prices.html. Accessed: 08/16/07

[3] Levin, M., Chottiner, J., Jaura, A., Ullman, Z., Votteler, J., “Design and Analysis of a Starter-Alternator Installation in a Hybrid-Electric Vehicle”, SAE Technical Paper 1999-01-0917

[4] Canova, M., Sevel, K., Guezennec, Y., Yurkovich, S., "Control of the Start/Stop of a Diesel Engine in a Parallel HEV: Modeling and Experiments", Proceeding of the ASME International Mechanical Engineering Congress and Exposition, 2006.

[5] Dorf, R., Bishop, R. Modern Control Systems. Tenth Edition. ©2005,2001 Pearson Education, Inc, New York.

[6] Kamen, E., Heck, B. Fundamentals of Signals and Systems. Second Edition. ©2000,1997 Prentice Hall, Inc, New Jersey.

[7] Canova, M., Sevel, K., Guezennec, Y., Yurkovich, S., "Control of the Start/Stop of a Diesel Engine in a Parallel HEV with a Belted Starter/Alternator", 7th International Conference on Engines for Automobile, 2007.

Appendix A – Embedded Algorithm MATLAB code

```
%A script which will attempt to find the specific minima for a given
%controller inside a given space.

%Controller: xxx, this script is "universal"
%Space: A set of values needs to be changed w/in the function to
define the
%space by which points will be tested

global minima_map

%Initializations
Vec1 = ;%specify an first parameter vector
Vec2 = ;%specify a second
i = 0;

%loop
n=0;m=0;h=0;
for m=1:length(Vec1)
    C1=Vec1(m);         %vary first parameter
    for h=1:length(Vec2)
        C2=Vec2(h);     %vary second
        x0=[C1 C2];     %specify start point
        i=i+1;          %increment holder
        [loc, min]=fminsearch(@startstoplqr,x0);  %find local minimum
        minima_map_over1(i,1:3)=loc;
        minima_map_over1(i,4)=min;
    end
end

%This is the embedded algorithm. The global algorithm is more of a
%user-process where the space is manipulated to try to center on a
global
%minima
```